

Simatic 505 Ethernet Device Driver Help

© 2008 Kepware Technologies

Table of Contents

1	Getting Started.....	3
	Help Contents.....	3
	Overview	3
2	Device Setup.....	3
	Device Setup	3
	Addressing Options.....	4
	Communications Parameters.....	5
	Cable Diagrams.....	6
3	Optimizing Ethernet Communications.....	7
	Optimizing Simatic505 Ethernet Communications.....	7
4	Data Types	8
	Data Types Description.....	8
5	Addressing	9
	Address Descriptions.....	9
	505-CP2572 Addressing.....	9
	505-CP1434-TCP Addressing.....	9
	Common Data Type Addressing.....	9
	Alarm Addressing.....	13
	Alarm Addressing.....	13
	AVF Bit Definitions.....	15
	ACF Bit Definitions.....	15
	Loop Addressing.....	16
	Loop Addressing.....	16
	LVF Bit Definitions.....	18
	LCF Bit Definitions.....	19
	LRSF Bit Definitions.....	20
	Find Forced Addressing.....	21
	Find Forced Addressing.....	21
6	Error Descriptions.....	22
	Error Descriptions.....	22
	Address Validation.....	23
	Address Validation.....	23
	Missing address.....	23
	Device address '<address>' contains a syntax error.....	23
	Address '<address>' is out of range for the specified device or register.....	23
	Data Type '<type>' is not valid for device address '<address>'.....	24
	Device address '<address>' is Read Only.....	24
	Array size is out of range for address '<address>'.....	24
	Array support is not available for the specified address: '<address>'.....	24
	Device Status Messages.....	25
	Device Status Messages.....	25
	Device '<device name>' is not responding.....	25
	Unable to write to '<address>' on device '<device name>'.....	25
	Driver Error Messages.....	25
	Driver Error Messages.....	25
	Winsock initialization failed (OS Error.....	25
	Winsock V1.1 or higher must be installed to use the Simatic 505 Ethernet device driver.....	26

Device '<device name>' input queue is full. The module is receiving requests faster than it.....	26
Device '<device name>' cannot execute a read request ('<address>', '<size>').....	26
Device '<device name>' cannot execute a write request ('<address>', '<size>').....	26
Device '<device name>' responded with Extended error '<error>' (Tag '<address>', Size '<size>').....	26
Device '<device name>' detected a NITP Protocol Error (Tag '<address>', Size '<size>').....	27
Device '<device name>' detected a Task Code error '<error>' (Tag '<address>', Size '<size>').....	27
Device '<Device name>', Write Value not in range1-->65536. (Tag 'FFSTART').....	27
7 Error Codes.....	27
Task Error Codes.....	27
 Index	 29

Simatic 505 Ethernet Device Driver Help

Help version 1.017

CONTENTS

[Overview](#)

What is the Simatic 505 Ethernet Device Driver?

[Device Setup](#)

How do I configure a device for use with this driver?

[Optimizing Your Simatic 505 Ethernet Communications](#)

How do I get the best performance from the Simatic 505 Ethernet driver?

[Data Types](#)

What data types does this driver support?

[Addressing](#)

How do I address a data location on a Simatic 505 device?

[Error Descriptions](#)

What error messages does the Simatic 505 Ethernet driver produce?

Overview

The Simatic 505 Ethernet Device Driver was designed specifically to be used in conjunction with 32 bit OPC Server products. It is intended for use in communicating with Simatic 505 Programmable Logic Controllers that may be accessed via either the Siemens Simatic 505-CP2572 Ethernet module, the Siemens 505-CP1434-TCP Card or the Control Technology Inc. 505-CP2572 card. The driver supports both TCP/IP and UDP transport protocols while using both CAMP and CAMP Packed Task Code messaging protocols for efficient data transfer.

Note: Please refer to the OPC Server Help documentation for operating system requirements.

Device Setup

Supported Devices

CTI Interface Cards

505-CP2572 or 505-CP2572A Ethernet Interface Cards on TI series CPUs TI545, 555 and 565. Simultaneous device connections are allowed. For more information, refer to [Connection Limitations](#) below.

Siemens Interface Cards

505-CP1434-TCP Ethernet Interface Card on TI Series CPUs TI545 and 565. Simultaneous device connections are allowed. For more information, refer to the *Siemens 505-CP1434-TCP Users Guide*.

CTI 2500 Controller (all models) Local Ethernet Port

The CTI 2500 Controller's port supports up to 3 TCP connections (no UDP). Users can connect using the TCP port 4450 or 1505. Set model to: 505-CP2572.

Supported Protocols

505: CAMP and CAMP Packed Task Code
IP: UDP and TCP/IP

Connection Timeout

This parameter specifies the time that the driver will wait for a connection to be made with a device. Depending on network load, the connect time may vary with each connection attempt. The default setting is 3 seconds. The valid range is 1 to 30 seconds.

Request Timeout

This parameter specifies the time that the driver will wait on a response from the device before giving up and going on to the next request. Longer timeouts only affect performance if a device is not responding. The default setting is 1000 milliseconds. The valid range is 100 to 30000 milliseconds.

Retry Attempts

This parameter specifies the number of times the driver will retry a message before giving up and going on to the next message. The default setting is 3 retries. The valid range is 1 to 10.

Device IDs

Up to 2048 devices may be defined on a given channel. Each device on the channel must be uniquely identified by its own IP address.

Connection Limitations

1. The default number of TCP connections allowed to the 2572 is 8. It is configurable in the Start Network Server command when configured from the PLC. The value can be set to a lower or higher number. In the old 2572, this is only configurable when started from PLC logic. In the 2572-A, this can also be configured when auto started from EEPROM.
2. One of the configuration DIP switches that enables the Data Share feature. When enabled, it automatically limits the number of TCP server connections to 2.
3. To connect to the PLC via the 505-CP1434-TCP card, configure a UDP Server Job in the card.

Addressing Options

0/1-Based Bit Addressing

Memory types that allow bit within Word (for example, V) can be referenced as a Boolean. The addressing notations for doing this are as follows:

```
<memory type><address>.<bit>
<memory type><address>:<bit>
<memory type><address>,<bit>
```

For each of these syntax, <bit> represents the bit number within the Word or DWord, depending on the memory type. 0/1-Based Bit Addressing provides two ways of addressing a bit within the given Word or DWord; 0-Based and 1-Based. 0-Based addressing simply means the "first bit" begins at 0. With 1-Based, the "first bit" begins at 1. The bit order for the Word or DWord is irrelevant with this option. In other words, it doesn't matter whether the "first bit" is the Most Significant Bit or the Least Significant Bit.

Note: In this driver, the first bit will either be bit 0 or bit 1 depending on this 0/1-Based Bit Addressing setting.

0-Based (Default Setting)

Data Type	Bit Range
Word	Bits 0–15
DWord	Bits 0–31

1-Based

Data Type	Bit Range
Word	Bits 1–16
DWord	Bits 1–32

Note: 0/1-Based Bit Addressing does not apply to non-bit addresses such as Word addresses in V memory. These addresses are always 1-Based and are not configurable.

Bit Order for V, K, WX, WY and STW

This option is used to select the order in which bits will be presented to V, K, WX, WY and STW memory types when bit-accessed.

Bit Order for Loops/Alarms

This option is used to select the order in which bits will be presented to Loop and Alarm memory types that are bit-accessible and are being bit-accessed (i.e. LCF)

MSB=Most Significant Bit
LSB=Least Significant Bit

Note: For the following example, the 1st through 16th bit signifies either 0-15 bits or 1-16 bits depending on if the driver is set at 0-Based Bit Addressing or 1-Based.

DWord follows the same bit order logic as Words except instead of 16 bits, there are 32 bits.

Bit 0 Is MSB of Word

MSB								LSB							
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

Bit 1 Is LSB (Default Setting) of Word

MSB								LSB							
16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1

Note for CTI2572 Users: Bit 0/1 Is MSB corresponds to the TICVU I/O Server and is not the default setting with this OPC Server. Select this option if applicable.

Bit 0/1 Is LSB corresponds to the TI Direct I/O Server (TIDIR). Select this option in order to make the OPC Server compatible with the TIDIR I/O Server. For instance, use the TIDIR I/O Server as a backup in certain applications.

Communications Parameters

Port Number

This parameter specifies the port number the remote device is configured to use. The default port number is 1505.

IP Protocol

This parameter specifies whether the driver should connect to the remote device using the User Datagram Protocol (UDP) or Transfer Control Protocol (TCP). The driver requires Winsock V1.1 or higher.

Note: If the Simatic 505-CP1434-TCP card is being used, then a UDP server job must also be set up if using the UDP IP Protocol (or a TCP server job if using the TCP IP Protocol). Using the TCP IP Protocol with retries set to value greater than the default of 3 is recommended.

Request Size

Request size refers to the number of bytes that may be requested from a device at one time. To refine the performance of this driver, the request size may be configured to one of the following settings: 32, 64, 128, 250 bytes. The default value is 250 bytes.

505 Protocol

This parameter specifies whether the driver should use *CAMP* or *CAMP Plus Packed Task Code* Protocol when communicating with device. When set to *CAMP*, the driver will use **only** the CAMP Memory Transfer protocol when communicating with device. When set to *CAMP Plus Packed Task Code*, the driver will use Packed Task Code, when possible and appropriate, in addition to CAMP Memory Transfer to communicate with device. The default value is *CAMP Plus Packed Task Code*.

TI565

Select the TI565 checkbox if using this PLC. The default value is unchecked. For more information, refer to [Common Data Type Addressing](#) Notes 5 and 6.

Note 1: Some address types, Strings and Arrays are not supported under *Packed Task Code* Protocol. **See Also:** [Address Descriptions](#).

Note 2: For best performance when using *CAMP Plus Packed Task Code*, the number of task codes per scan on the PLC should be set to the maximum available (typically 8). If using TISOFT, this can be set by using Aux Function 19.

CAMP vs. Packed Task Code

For applications where the data can be divided into large blocks, using CAMP memory transfer will result in the most

efficient data transfer. Note that, in one CAMP read, up to 125 words could be returned which could include 2000 discrete points. However, since CAMP memory transfer allows only one data type and only one memory range per message, applications which read many small blocks or many different data types may benefit from using the Packed Task Code format.

For example, reading 10 V memory values, 10 WY values, 30 WX values and 2 loop process variables would require 4 messages (one for the V, one for the WY, one for the WX and one for the loop). Using the packed task code format (which allows 14 NITP task code requests to be placed in a single message), allows data to be obtained in a single message.

Optimum Performance Recommendations

All devices on a channel should be set to the same 505 Protocol mode. When possible, the project's data should be divided into the following categories:

1. Items that are capable of utilizing Packed Task Code.
2. Items that are not capable of utilizing Packed Task Code. This includes the following:
 - Arrays
 - Strings
 - Address types that are not supported with Packed Task Code (For example: DCP)
 - Contiguous address ranges > 36 Words. (For example: V1, V2, V3, ... V36, V37, V38)

All items in Category 1 should be placed into a device that is set to *CAMP Plus Packed Task Code* 505 Protocol mode on a different channel from Category 2 items. We recommend that the Request size be set to 64 or 128.

All items in Category 2 should be placed into a device that is set to CAMP 505 Protocol mode on a different channel from Category 1 items. We recommend that the Request size be set to 250.

Note: When reading a project file prior to addition of 505 Protocol option, the default value is set to CAMP.

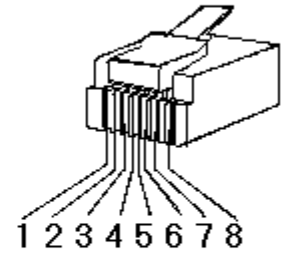
Cable Diagrams

Patch Cable (Straight Through)

TD +	1	OR/WHT	OR/WHT	1	TD +
TD -	2	OR	OR	2	TD -
RD +	3	GRN/WHT	GRN/WHT	3	RD +
	4	BLU	BLU	4	
	5	BLU/WHT	BLU/WHT	5	
RD -	6	GRN	GRN	6	RD -
	7	BRN/WHT	BRN/WHT	7	
	8	BRN	BRN	8	

RJ45 RJ45

10 BaseT



Crossover Cable

TD +	1	OR/WHT	GRN/WHT	1	TD +
TD -	2	OR	GRN	2	TD -
RD +	3	GRN/WHT	OR/WHT	3	RD +
	4	BLU	BLU	4	
	5	BLU/WHT	BLU/WHT	5	
RD -	6	GRN	OR	6	RD -
	7	BRN/WHT	BRN/WHT	7	
	8	BRN	BRN	8	

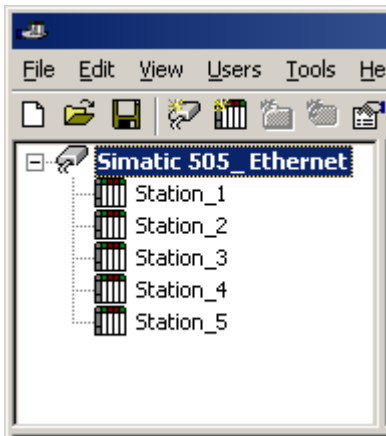
RJ45 RJ45

8-pin RJ45

Optimizing Simatic 505 Ethernet Communications

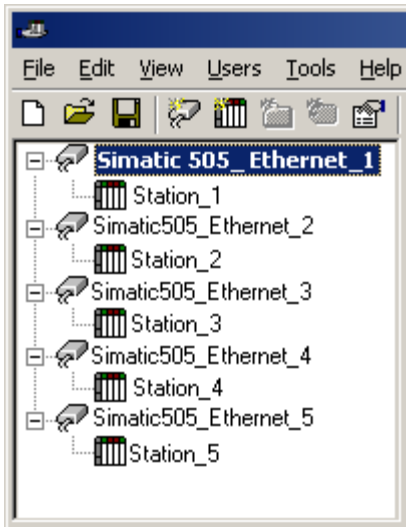
The Simatic 505 Ethernet driver has been designed to provide the best performance with the least amount of impact on the system's overall performance. While the Simatic 505 Ethernet driver is fast, there are a couple of guidelines that can be used in order to control and optimize the application and gain maximum performance.

Our server refers to communications protocols like Simatic 505 Ethernet as a channel. Each channel defined in the application represents a separate path of execution in the server. Once a channel has been defined, a series of devices must then be defined under that channel. Each of these devices represents a single Simatic 505 controller from which data will be collected. While this approach to defining the application will provide a high level of performance, it won't take full advantage of the Simatic 505 Ethernet driver or the network. An example of how the application may appear when configured using a single channel is shown below.



Each device appears under a single Simatic 505 Ethernet channel. In this configuration, the driver must move from one device to the next as quickly as possible in order to gather information at an effective rate. As more devices are added or more information is requested from a single device, the overall update rate begins to suffer.

If the Simatic 505 Ethernet driver could only define one single channel, then the example shown above would be the only option available; however, the Simatic 505 Ethernet driver can define up to 100 channels. Using multiple channels distributes the data collection workload by simultaneously issuing multiple requests to the network. An example of how the same application may appear when configured using multiple channels to improve performance is shown below.



Each device has now been defined under its own channel. In this new configuration, a single path of execution is dedicated to the task of gathering data from each device. If the application has 100 or fewer devices, it can be optimized exactly how it is shown here.

The performance will improve even if the application has more than 100 devices. While 100 or fewer devices may be ideal, the application will still benefit from additional channels. Although by spreading the device load across all channels will cause the server to move from device to device again, it can now do so with far less devices to process on a single channel.

Block Size, which is available on each defined device, can also affect the Simatic 505 Ethernet driver's performance. Block Size refers to the number of bytes that may be requested from a device at one time. To refine the performance of this driver, configure Block Size to one of the following settings: 32, 64, 128, or 250 bytes. Depending on the Simatic 505 Ethernet device model, the Block Size setting affect the application's performance drastically. A default value of 250 bytes is recommended. If an application has large requests for consecutively ordered data, however, block size should be increased.

Data Types Description

Data Type	Description
Boolean	Single bit
Word	Unsigned 16 bit value bit 0 is the low bit bit 15 is the high bit
Short	Signed 16 bit value bit 0 is the low bit bit 14 is the high bit

	bit 15 is the sign bit
DWord	Unsigned 32 bit value bit 0 is the low bit bit 31 is the high bit
Long*	Signed 32 bit value bit 0 is the low bit bit 30 is the high bit bit 31 is the sign bit
Float**	32 bit floating point value The driver interprets two consecutive registers as a floating-point value by making the second register the high word and the first register the low word.
String	Null terminated ASCII string Includes Hi-Lo Lo-Hi byte order selection.

*Long is the same as Double in the TISOFT programming software.

**Float is the same as Real in the TISOFT programming software.

Address Descriptions

Address specifications vary depending on the model in use. Select a link from the following list to obtain specific address information for the model of interest.

[505-CP2572](#)

[505-CP1434-TCP](#)

505-CP2572 Addressing

The following sections define addressing for the CP2572 card. Refer to the Event Log section within the Server Options chapter of the server help file for detailed information on how the event logger works.

[Common Data Type Addressing](#)

[Alarm Addressing](#)

[Loop Addressing](#)

[Find Forced Addressing](#)

505-CP1434-TCP Addressing

The following sections define addressing for the CP1434 card. Refer to the Event Log section within the Server Options chapter of the server help file for detailed information on how the event logger works.

[Common Data Type Addressing](#)

[Alarm Addressing](#)

[Loop Addressing](#)

[Find Forced Addressing](#)

Common Data Type Addressing

The driver supports the following addresses. The default data type for each address type is indicated in **bold**.

Address Type	Format	Range	Data Types	Access
Discrete Input	X<address>	1-65536	Boolean	Read/Write
Discrete Output	Y<address>	1-65536	Boolean	Read/Write
Word Input	WX<address>	1-65536	Short, Word	Read/Write

		1-65535	Long, DWord, Float	
Word Input Signed Integer	WX<address>S	1-65536 1-65535	Short , Word Long, DWord, Float	Read/Write
Word Input Bit Access	WX<address>.<bit> WX<address>:<bit> WX<address>,<bit>	1-65536 Bit 0/1-15/16 0/1-Based Bit Addressing	Boolean, Short, Word	Read/Write
Word Output	WY<address>	1-65536 1-65535	Short, Word Long, DWord, Float	Read/Write
Word Output Signed Integer	WY<address>S	1-65536 1-65535	Short , Word Long, DWord, Float	Read/Write
Word Output Bit Access	WY<address>.<bit> WY<address>:<bit> WY<address>,<bit>	1-65536 Bit 0/1-15/16 0/1-Based Bit Addressing	Boolean, Short, Word	Read/Write
Discrete Control (Internal Coil) (See Note 5)	C<address> CR<address>	1-65536	Boolean	Read/Write
V-Word Memory	V<address>	1-16777215 1-16777214	Short, Word , BCD Long, DWord, Float	Read/Write
V-Word Memory Signed Integer	V<address>S	1-16777215 1-16777214	Short , Word, BCD Long, DWord, Float	Read/Write
V-Word Memory Double Word	V<address>D	1-16777215 1-16777214	Short, Word, BCD Long, DWord , Float	Read/Write
V-Word Memory BCD Decimal	V<address>B	1-16777215 1-16777214	Short, Word, BCD Long, DWord, Float	Read/Write
V-Word Memory Floating Point	V<address>R V<address>P V<address>.	1-16777215 1-16777214	Short, Word, BCD Long, DWord, Float	Read/Write
V-Word Memory Bit Access	V<address>.<bit> V<address>:<bit> V<address>,<bit>	1-16777215 Bit 0/1-15/16 0/1-Based Bit Addressing	Boolean, Short, Word	Read/Write
V-Word Memory As Standard String with HiLo Byte Order (See Note 4)	V<address>.<length>H	V1.2H-V16777090.250H .Bit is string length. Range 2 to 250 bytes.	String	Read/Write
V-Word Memory As Standard String with LoHi Byte Order (See Note 4)	V<address>.<length>L	V1.2L-V16777090.250L .Bit is string length Range 2 to 250 bytes.	String	Read/Write
V-Word Memory As Null Terminated String with LoHi Byte Order (See Note 4)	V<first>-<last>C <first> is first word <last> is last word	1-16777215 (last-first)=string length Range 1 to 125 words.	String	Read/Write
V-Word Memory As String with LoHi Byte Order with Length Byte (See Note 4)	V<first>-<last>P <first> is first word <last> is last word	1-16777215 (last - first)=string length Range 1 to 125 words.	String	Read/Write
V-Word Memory As String with LoHi Byte Order Blank-Padded on Right (See Note 4)	V<first>-<last> <first> is first word <last> is last word	1-16777215 (last-first)=string length Range 1 to 125 words.	String	Read/Write
Constant Memory	K<address>	1-16777215 1-16777214	Short, Word , BCD Long, DWord, Float	Read/Write
Constant Memory Signed Integer	K<address>S	1-16777215 1-16777214	Short , Word, BCD Long, DWord, Float	Read/Write
Constant Memory Double Word	K<address>D	1-16777215 1-16777214	Short, Word, BCD Long, DWord , Float	Read/Write

Constant Memory BCD Decimal	K<address>B	1-16777215 1-16777214	Short, Word, BCD Long, DWord, Float	Read/Write
Constant Memory Floating Point	K<address>R K<address>P K<address>.	1-16777215 1-16777214	Short, Word, BCD Long, DWord, Float	Read/Write
Constant Memory Bit Access	K<address>.<bit> K<address>:<bit> K<address>,<bit>	1-16777215 Bit 0/1-15/16 0/1-Based Bit Addressing	Boolean, Short, Word	Read/Write
System Status	STW<address>	1-65536 1-65535	Short, Word Long, DWord, Float	Read Only
System Status Bit Access	STW<address>.<bit> STW<address>:<bit> STW<address>,<bit>	1-65536 Bit 0/1-15/16 0/1-Based Bit Addressing	Boolean, Short, Word	Read/Write
Timer/Counter Preset	TCP<address>	1-65536 1-65535	Short, Word Long, DWord, Float	Read/Write
Timer/Counter Current	TCC<address>	1-65536 1-65535	Short, Word Long, DWord, Float	Read/Write
Drum Step Preset	DSP<address>	1-65536 1-65535	Short, Word Long, DWord, Float	Read/Write
Drum Step Current	DSC<address>	1-65536 1-65535	Short, Word Long, DWord, Float	Read/Write
Drum Count Preset (See Note 3)	DCP<drum>.<step> DCP<drum>:<step> DCP<drum>,<step>	drum 1-65536 step 1-16	Short, Word	Read/Write
Drum Current Count	DCC<address>	1-65536 1-65535	Short, Word Long, DWord , Float	Read Only

Note: The actual number of addresses available for of each type is dependent on the configuration of the PLC. If at runtime the driver finds that an address is not present in the device, the driver will post an error message and remove the tag from its scan list.

Array Support

Non-Boolean addresses accept array notation. To view data as an array, append '[rows][cols]' to an address (e.g., V500 [2][2], V100 [12]).

Note: DCP registers have to be accessed in groups of 16 words. Therefore, when writing to a DCP register, the driver performs a read to obtain the current values of all 16-word values (DCP00001.01 - DCP00001.16). The driver modifies the one register that is being written to and sends the 16 values to the device. Since the driver performs this read/modify/write procedure, it could be possible for any one of the registers to change between the read and write transaction. The result would be a new value being written over by an older value. A recommendation would be to write to another type of memory location (e.g., Vxxxxx) and using ladder, transfer the value to a DCP register.

String Support

Standard String

The Simatic 505 Ethernet driver supports reading and writing V registers as an ASCII string. When using V registers for string data, each register will contain two bytes of ASCII data. The order of the ASCII data within a given register can be selected when the string is defined. The length of the string can be from 2 to 250 bytes and is entered in place of a bit number. The length must be entered as an even number. The byte order is specified by appending either a "H" or "L" to the address.

Null Terminated

This type of string is a Standard String with LoHi byte order that is null terminated at all times. This means the last byte will always be a null terminator, essentially losing one character place when compared to a Standard String with LoHi byte order.

Length + String

This type of string is also a Standard String with LoHi byte order except the MSB of the first word contains the desired length of the string to follow. This means one character place is lost when compared to a Standard String with LoHi byte order. Based on this length, a string is formed up to the max size set by the address range (ie. (<last word> - <first word>) / 2 - 1 bytes). Thus the max length of the string can vary between the length byte value and (<last word>-<first word>) / 2 - 1 bytes. Note that any null terminators in the string can cut the length short of these max

lengths. If a string to be written is less than this max length, it will be padded on the right with blank-spaces until all max bytes contain a non-NULL value.

Blank-Padded on Right

This type of string is a Standard String with LoHi byte order except the string length is always the max because the string is padded with blank spaces on the right until its length becomes the max (no NULLs). Recall the max length of the string is $(\text{<last word> - <first word>}) / 2$ bytes in the address. If a string to be written is less than this max length, it too will be padded on the right with blank-spaces until all max bytes contain a non-NULL value.

Example 1: Standard String

To address a string starting at V200 with a length of 50 bytes and HiLo byte order enter:
V200.50H

Example 2: Standard String

To address a string starting at V500 with a length of 38 bytes and LoHi byte order enter:
V500.38L

Example 3: Null Terminated

Given:

```
V200/MSB = 'h'  
V200/LSB = 'e'  
V201/MSB = 'l'  
V201/LSB = 'l'  
V202/MSB = 'o'  
V202/LSB = ' '  
V203/MSB = 'w'  
V203/LSB = 'o'  
V204/MSB = 'r'  
V204/LSB = 'l'  
V205/MSB = 'd'
```

To address a null terminated string starting at V200 with a length of 10 bytes and the memory data is as stated above:
V200-204C would read "hello wor".

Notice how only 9 characters can be displayed since the last is the null terminator

Example 4: Length Byte + String

Given:

```
V200/MSB = 0x04  
V200/LSB = 'e'  
V201/MSB = 'l'  
V201/LSB = 'l'  
V202/MSB = 'o'  
V202/LSB = ' '  
V203/MSB = 'w'  
V203/LSB = 'o'  
V204/MSB = 'r'  
V204/LSB = 'l'  
V205/MSB = 'd'
```

To address a string with length byte information, starting at V200 and the memory data is as stated above:
V200-204P would read "ello"

The length byte is 0x04 so only 4 characters are displayed

If the length byte were 0x07, V200-204P would read "ello wo"

Example 5: Blank-Padded on Right

Given:

```
V200/MSB = 'h'  
V200/LSB = 'e'  
V201/MSB = 'l'  
V201/LSB = 'l'  
V202/MSB = 'o'  
V202/LSB = NULL  
V203/MSB = 'w'  
V203/LSB = 'o'  
V204/MSB = 'r'  
V204/LSB = 'l'
```

V205/MSB = 'd'

To address a blank padded string starting at V200 with a length of 10 bytes and the memory data is as stated above: V200-204 would read "hello ".

Notice how all 10 characters are displayed with the null terminated string "hello" padded with 5 spaces to fill the 5 words.

If we were to write "world" to this address the memory data would look as follows:

```
V200/MSB = 'w'
V200/LSB = 'o'
V201/MSB = 'r'
V201/LSB = 'l'
V202/MSB = 'd'
V202/LSB = ' '
V203/MSB = ' '
V203/LSB = ' '
V204/MSB = ' '
V204/LSB = ' '
V205/MSB = ' '
V205/LSB = ' '
```

Since "world" is less than 10 characters in length, it is padded with blanks to become 10 characters in length.

Note 1: C and CR memory types are supported on the 565 model controller by setting the TI565 checkbox in [Communications Parameters](#).

Note 2: The following address type(s), Strings and Arrays are not supported under Packed Task Code Protocol. To increase efficiency, CAMP Protocol will be used instead.

Drum Count Preset (DCP)
 Discrete Input (X)
 Discrete Output (Y)
 Discrete Control (C or CR)
 Addresses greater than '30720' for Drum Step Preset (DSP) and Drum Step Current (DSC)

If the TI565 option is checked in [Communications Parameters](#), however, Packed Task Code Protocol will be used for address types C, X and Y.

Alarm Addressing

The driver supports the following addresses. The default data type for each address type is indicated in **bold**.

Address Type	Format	Range	Data Types	Access
Alarm Status (V) Flags	AVF<address>	1-65536	Word	Read/Write
Alarm Status (V) Flag Bit Access AVF Bit Definitions	AVF<address>.<bit> AVF<address>:<bit> AVF<address>,<bit>	1-65536 Bit 0/1-15/16 0/1-Based Bit Addressing	Boolean, Word	Read/Write
Alarm Control (C)Flags	ACF<address>	1-65535	DWord	Read/Write
Alarm Control (C) Flag Bit Access ACF Bit Definitions	ACF<address>.<bit> ACF<address>:<bit> ACF<address>,<bit>	1-65535 Bit 0/1-31/32 0/1-Based Bit Addressing	Boolean, DWord	Read/Write
Alarm PV High Alarm	APVH<address>	1-65535	Float	Read/Write
Alarm PV Low Alarm	APVL<address>	1-65535	Float	Read/Write
Alarm Process Variable	APV<address>	1-65535	Float	Read/Write
Alarm High Limit	AHA<address>	1-65535	Float	Read/Write
Alarm Low Limit	ALA<address>	1-65535	Float	Read/Write
Alarm Orange Deviation Alarm	AODA<address>	1-65535	Float	Read/Write
Alarm Yellow Deviation Alarm	AYDA<address>	1-65535	Float	Read/Write

Alarm Sample Rate	ATS<address>	1-65535	Float	Read/Write
Alarm Setpoint	ASP<address>	1-65535	Float	Read/Write
Alarm Error	AERR<address>	1-65535	Float	Read Only
Alarm High High Alarm Limit	AHHA<address>	1-65535	Float	Read/Write
Alarm Low Low Alarm Limit	ALLA<address>	1-65535	Float	Read/Write
Alarm Rate of Change Alarm	ARCA<address>	1-65535	Float	Read/Write
Alarm Setpoint High Limit	ASPH<address>	1-65535	Float	Read/Write
Alarm Setpoint Low Limit	ASPL<address>	1-65535	Float	Read/Write
Alarm Alarm Deadband	AADB<address>	1-65535	Float	Read/Write
Alarm Raw High Alarm Limit	AHAR<address>	1-65536	Word	Read/Write
Alarm Raw Low Alarm Limit	ALAR<address>	1-65536	Word	Read/Write
Alarm Raw Process Variable	APVR<address>	1-65536	Word	Read/Write
Alarm Raw Orange Deviation	AODAR<address>	1-65536	Word	Read/Write
Alarm Raw Yellow Deviation	AYDAR<address>	1-65536	Word	Read/Write
Alarm Raw Setpoint	ASPR<address>	1-65536	Word	Read/Write
Alarm Raw Alarm Deadband	ADBR<address>	1-65536	Word	Read/Write
Alarm Raw Error	AERRR<address>	1-65536	Word	Read Only
Alarm Raw High-High Alarm Limit	AHHAR<address>	1-65536	Word	Read/Write
Alarm Raw Low-Low Alarm Limit	ALLAR<address>	1-65536	Word	Read/Write
Alarm Raw Setpoint Low Limit	ASPLR<address>	1-65536	Word	Read/Write
Alarm Raw Setpoint High Limit	ASPHR<address>	1-65536	Word	Read/Write
Alarm MSW Alarm C Flags	ACFH<address>	1-65536	Word	Read/Write
Alarm LSW Alarm C Flags	ACFL<address>	1-65536	Word	Read/Write
Alarm ACK Flag	AACK<address>	1-65536	Word	Read Only

Note 1: The actual number of addresses available for of each type is dependent on the configuration of the PLC. If at runtime the driver finds that an address is not present in the device, the driver will post an error message and remove the tag from its scan list.

Note 2: Array Support: Non-Boolean addresses accept array notation. To view data as an array, append '[rows][cols]' to an address (e.g., V500 [2][2], V100 [12]).

Note 3: The following address type(s) and Arrays are not supported under Packed Task Code Protocol. CAMP Protocol will be used instead.

Alarm Control (C)Flags (ACF)
 Alarm Raw High Alarm Limit (AHAR)
 Alarm Raw Low Alarm Limit (ALAR)

Alarm Raw Process Variable (APVR)
 Alarm Raw Orange Deviation (AODAR)
 Alarm Raw Yellow Deviation (AYDAR)
 Alarm Raw Setpoint (ASPR)
 Alarm Raw Alarm Deadband (ADBR)
 Alarm Raw Error (AERRR)
 Alarm Raw High-High Alarm Limit (AHHAR)
 Alarm Raw Low-Low Alarm Limit (ALLAR)
 Alarm Raw Setpoint Low Limit (ASPLR)
 Alarm Raw Setpoint High Limit (ASPHR)

AVF Bit Definitions

Analog Alarm Variable Flag Data Element Format

MSB		LSB	
ABCD	EFGH	IJKL	MNOP

The AVF bit has the following format:

V/W, X/Y

V=0-based bit addressing, bit 0 is MSB

W=1-based bit addressing, bit 1 is MSB (Simatic 505 Documentation)

X=0-based bit addressing, bit 0 is LSB (Server Default Setting)

Y=1-based bit addressing, bit 1 is LSB

Element	AVF bit	Description
A	0/1, 15 /16	1=Enable alarm**
B	1/2, 14 /15	1=Disable alarm**
C	2/3, 13 /14	1=PV is in high-high alarm
D	3/4, 12 /13	1=PV is in high alarm
E	4/5, 11 /12	1=PV is in low alarm
F	5/6, 10 /11	1=PV is in low-low alarm
G	6/7, 9 /10	1=PV is in yellow deviation alarm
H	7/8, 8 /9	1=PV is in orange deviation alarm
I	8/9, 7 /8	1=PV is in rate of change alarm
J	9/10, 6 /7	1=Broken transmitter alarm
K	10/11, 5 /6	1=Analog alarm is overrunning
L	11/12, 4 /5	1=Alarm is enabled *
M	12/13, 3 /4	Not used, set to 0
N	13/14, 2 /3	Not used, set to 0
O	14/15, 1 /2	Not used, set to 0
P	15/16, 0 /1	Not used, set to 0

*If a word is selected for the analog alarm V-flags, element L is written as bit 12 is written. If a C or Y is selected, bit 12 (element L) is not used.

**Elements A and B are write-only triggers. Writing 1 to the triggers sets the mode. The mode (status) is reported in Element L (Bit 12).

Note: If an analog alarm is programmed and not disabled, the controller will begin to monitor the programmed variables as soon as the controller is placed in RUN mode.

ACF Bit Definitions

Analog Alarm Control Flag Data Element Format

MSB	LSB
-----	-----

ABCD	EFGH	IJKL	MNOP	QRST	UVWX	YZab	cdef
------	------	------	------	------	------	------	------

The ACF bit has the following format:

V/W, **X**/Y

V=0-based bit addressing, bit 0 is MSB

W=1-based bit addressing, bit 1 is MSB (Simatic 505 Documentation)

X=0-based bit addressing, bit 0 is LSB (Server Default Setting)

Y=1-based bit addressing, bit 1 is LSB

Element	ACF bit	Description
A	0/1, 31 /32	0=PV scale 0% offset 1=PV scale 20% offset
B	1/2, 30 /31	1=Take square root of PV
C	2/3, 29 /30	1=Monitor HIGH/LOW alarms
D	3/4, 28 /29	1=Monitor HIGH-HIGH/LOW-LOW alarms
E	4/5, 27 /28	1=Monitor yellow/orange deviation alarm
F	5/6, 26 /27	1=Monitor rate-of-change alarm
G	6/7, 25 /26	1=Monitor broken transmitter alarm
H	7/8, 24 /25	0=Local setpoint 1=Remote setpoint
I	8/9, 23 /24	Unused, set to 0
J	9/10, 22 /23	Unused, set to 0
K	10/11, 21 /22	Unused, set to 0
L	11/12, 20 /21	Unused, set to 0
M	12/13, 19 /20	Unused, set to 0
N	13/14, 18 /19	Unused, set to 0
O	14/15, 17 /18	Unused, set to 0
P	15/16, 16 /17	Unused, set to 0
Q	16/17, 15 /16	Unused, set to 0
R	17/18, 14 /15	Unused, set to 0
S	18/19, 13 /14	Unused, set to 0
T	19/20, 12 /13	Unused, set to 0
U	20/21, 11 /12	0=Process variable is unipolar 1=Process variable is bipolar
V	21/22, 10 /11	Unused, set to 0
W	22/23, 9 /10	Contains SF program number (if an SF program is scheduled to be called)
X	23/24, 8 /9	Contains SF program number (if an SF program is scheduled to be called)
Y	24/25, 7 /8	Contains SF program number (if an SF program is scheduled to be called)
Z	25/26, 6 /7	Contains SF program number (if an SF program is scheduled to be called)
a	26/27, 5 /6	Contains SF program number (if an SF program is scheduled to be called)
b	27/28, 4 /5	Contains SF program number (if an SF program is scheduled to be called)
c	28/29, 3 /4	Contains SF program number (if an SF program is scheduled to be called)
d	29/30, 2 /3	Contains SF program number (if an SF program is scheduled to be called)
e	30/31, 1 /2	Contains SF program number (if an SF program is scheduled to be called)
f	31/32, 0 /1	Contains SF program number (if an SF program is scheduled to be called)

Loop Addressing

The driver supports the following addresses. The default data type for each address type is indicated in **bold**.

Address Type	Format	Range	Data Types	Access
--------------	--------	-------	------------	--------

Loop Gain	LKC<address>	1-65535	Float	Read/Write
Loop Reset Time (min)	LTI<address>	1-65535	Float	Read/Write
Loop Rate Time (min)	LTD<address>	1-65535	Float	Read/Write
Loop High Alarm Limit	LHA<address>	1-65535	Float	Read/Write
Loop Low Alarm Limit	LLA<address>	1-65535	Float	Read/Write
Loop Process Variable	LPV<address>	1-65535	Float	Read/Write
Loop PV High Limit	LPVH<address>	1-65535	Float	Read/Write
Loop PV Low Limit	LPVL<address>	1-65535	Float	Read/Write
Loop Orange Dev Alarm Limit	LODA<address>	1-65535	Float	Read/Write
Loop Yellow Dev Alarm Limit	LYDA<address>	1-65535	Float	Read/Write
Loop Sample Rate	LTS<address>	1-65535	Float	Read/Write
Loop Setpoint	LSP<address>	1-65535	Float	Read/Write
Loop Output (%)	LMN<address>	1-65535	Float	Read/Write
Loop Status (V) Flags	LVF<address>	1-65536	Word	Read/Write
Loop Status (V) Flag Bit Access LVF Bit Definitions	LVF<address>.<bit> LVF<address>:<bit> LVF<address>,<bit>	1-65536 Bit 0/1-15/16 0/1-Based Bit Addressing	Boolean, Word	Read/Write
Loop Control (C) Flags	LCF<address>	1-65535	DWord	Read/Write
Loop Control (C) Flag Bit Access LCF Bit Definitions	LCF<address>.<bit> LCF<address>:<bit> LCF<address>,<bit>	1-65535 Bit 0/1-31/32 0/1-Based Bit Addressing	Boolean, DWord	Read/Write
Loop Ramp/Soak Status Flags	LRSF<address>	1-65536	Word	Read/Write
Loop Ramp/Soak Status Flag Bit Access LRSF Bit Definitions	LRSF<address>.<bit> LRSF<address>:<bit> LRSF<address>,<bit>	1-65536 Bit 0/1-15/16 0/1-Based Bit Addressing	Boolean, Word	Read/Write
Loop Error	LERR<address>	1-65535	Float	Read Only
Loop Bias	LMX<address>	1-65535	Float	Read/Write
Loop High-High Alarm Limit	LHHA<address>	1-65535	Float	Read/Write
Loop Low-Low Alarm Limit	LLLA<address>	1-65535	Float	Read/Write
Loop Rate of Change Alarm Limit	LRCA<address>	1-65535	Float	Read/Write
Loop Setpoint High Limit	LSPH<address>	1-65535	Float	Read/Write
Loop Setpoint Low Limit	LSPL<address>	1-65535	Float	Read/Write
Loop Alarm Deadband	LADB<address>	1-65535	Float	Read/Write
Loop Raw High Alarm Limit	LHAR<address>	1-65536	Word	Read/Write
Loop Raw Low Alarm Limit	LLAR<address>	1-65536	Word	Read/Write
Loop Raw Process Variable	LPVR<address>	1-65536	Word	Read/Write
Loop Raw Orange Dev Alarm Limit	LODAR<address>	1-65536	Word	Read/Write
Loop Raw Yellow Dev Alarm Limit	LYDAR<address>	1-65536	Word	Read/Write
Loop Raw Output	LMNR<address>	1-65536	Word	Read/Write
Loop Raw Setpoint	LSPR<address>	1-65536	Word	Read/Write

Loop Raw Error	LERRR<address>	1-65536	Word	Read Only
Loop Raw High-High Alarm Limit	LHHAR<address>	1-65536	Word	Read/Write
Loop Raw Low-Low Alarm Limit	LLLAR<address>	1-65536	Word	Read/Write
Loop Raw Alarm Deadband	LADBR<address>	1-65536	Word	Read/Write
Loop Raw Bias	LMXR<address>	1-65536	Word	Read/Write
Loop Raw Setpoint Low Limit	LSPLR<address>	1-65536	Word	Read/Write
Loop Raw Setpoint High Limit	LSPHR<address>	1-65536	Word	Read/Write
Loop C Flags - MSW	LCFH<address>	1-65536	Word	Read/Write
Loop C Flags - LSW	LCFL<address>	1-65536	Word	Read/Write
Loop Derivative Gain Limit Coef.	LKD<address>	1-65535	Float	Read/Write
Loop Ramp/Soak Step Number	LRSN<address>	1-65536	Word	Read/Write
Loop Alarm Ack Flags	LACK<address>	1-65536	Word	Read Only

Note 1: The actual number of addresses available for of each type is dependent on the configuration of the PLC. If at runtime the driver finds that an address is not present in the device, the driver will post an error message and remove the tag from its scan list.

Note 2: Array Support: Non-Boolean addresses accept array notation. To view data as an array, append '[rows][cols]' to an address (e.g., V500 [2][2], V100 [12]).

Note 3: The following address type(s) and Arrays are not supported under Packed Task Code Protocol. CAMP Protocol will be used instead.

- Loop Control (C)Flags (LCF)
- Loop Raw High Alarm Limit (LHAR)
- Loop Raw Low Alarm Limit (LLAR)
- Loop Raw Process Variable (LPVR)
- Loop Raw Orange Deviation (LODAR)
- Loop Raw Yellow Deviation (LYDAR)
- Loop Raw Output (LMNR)
- Loop Raw Setpoint (LSPR)
- Loop Raw Error (LERRR)
- Loop Raw High-High Alarm Limit (LHHAR)
- Loop Raw Low-Low Alarm Limit (LLLAR)
- Loop Raw Alarm Deadband (LADBR)
- Loop Raw Bias (LMXR)
- Loop Raw Setpoint Low Limit (LSPLR)
- Loop Raw Setpoint High Limit (LSPHR)

LVF Bit Definitions

Loop Variable Flag Data Element Format

MSB		LSB	
ABCD	EFGH	IJKL	MNOP

The LVF bit has the following format:

- V/W, X/Y
- V=0-based bit addressing, bit 0 is MSB
- W=1-based bit addressing, bit 1 is MSB (Simatic 505 Documentation)
- X=0-based bit addressing, bit 0 is LSB (Server Default Setting)**
- Y=1-based bit addressing, bit 1 is LSB

Element	LVF bit	Description
A	0/1, 15 /16	1=Go to manual mode*
B	1/2, 14 /15	1=Go to auto mode*
C	2/3, 13 /14	1=Go to cascade mode*
D	3/4, 12 /13	<i>D E=description**</i>
E	4/5, 11 /12	0 0=Loop is in manual mode 1 0=Loop is in auto mode 0 1=Loop is in cascade mode
F	5/6, 10 /11	0=Error is positive 1=Error is negative
G	6/7, 9 /10	1=PV is in high-high alarm
H	7/8, 8 /9	1=PV is in high alarm
I	8/9, 7 /8	1=PV is in low alarm
J	9/10, 6 /7	1=PV is in low-low alarm
K	10/11, 5 /6	1=PV is in yellow deviation alarm
L	11/12, 4 /5	1=PV is in orange deviation alarm
M	12/13, 3 /4	1=PV is in rate-of-change alarm
N	13/14, 2 /3	1=Broken transmitter alarm
O	14/15, 1 /2	1=Loop is overrunning
P	15/16, 0 /1	Spare: Set to 0

*Elements A, B and C are write-only triggers. Write 1 to set the mode. The element will always show a value of 0 (zero).

**Elements D and E will indicate the mode that the loop is in. These elements are read only.

Note: Elements F through P are read only and indicate the error and alarm state that the loop is in.

LCF Bit Definitions

Loop Control Flag Data Element Format

MSB				LSB			
ABCD	EFGH	IJKL	MNOP	QRST	UVWX	YZab	cdef

The LCF bit has the following format:

V/W, **X**/Y

V=0-based bit addressing, bit 0 is MSB

W=1-based bit addressing, bit 1 is MSB (Simatic 505 Documentation)

X=0-based bit addressing, bit 0 is LSB (Server Default Setting)

Y=1-based bit addressing, bit 1 is LSB

Element	LCF bit	Description
A	0/1, 31 /32	0=PV scale 0% offset 1=PV scale 20% offset - only valid if PV is unipolar (see element U)
B	1/2, 30 /31	1=Take square root of PV
C	2/3, 29 /30	1=Monitor HIGH/LOW alarms
D	3/4 28 /29	1=Monitor HIGH-HIGH/LOW-LOW alarms
E	4/5, 27 /28	1=Monitor yellow/orange deviation alarm
F	5/6, 26 /27	1=Monitor rate-of-change alarm
G	6/7, 25 /26	1=Monitor broken transmitter alarm
H	7/8, 24 /25	PID algorithm type 0=Position algorithm 1=Velocity algorithm
I	8/9, 23 /24	0=Direct acting

		1=Reverse acting
J	9/10, 22/23	1=Control based on error squared
K	10/11, 21/22	1=Control based on error deadband
L	11/12, 20/21	1=Auto-mode lock
M	12/13, 19/20	1=Cascade-mode lock
N	13/14, 18/19	1=Setpoint lock
O	14/15, 17/18	0=Output scale 0% offset 1=Output scale 20% offset - only valid if output is unipolar (see element T)
P	15/16, 16/17	P Q
Q	16/17, 15/16	0 1=No special function 1 0=Special function on the process variable 0 1=Special function on the setpoint 1 1=Special function on the output
R	17/18, 14/15	1=Freeze bias when output is out of range
S	18/19, 13/14	1=Ramp/Soak on the setpoint
T	19/20, 12/13	0=Output is unipolar 1=Output is bipolar
U	20/21, 11/12	0=PV is unipolar 1=PV is bipolar
V	21/22, 10/11	1=Perform derivative gain limiting
W	22/23, 9/10	Contains SF program number (if an SF program is scheduled to be called)
X	23/24, 8/9	Contains SF program number (if an SF program is scheduled to be called)
Y	24/25, 7/8	Contains SF program number (if an SF program is scheduled to be called)
Z	25/26, 6/7	Contains SF program number (if an SF program is scheduled to be called)
a	26/27, 5/6	Contains SF program number (if an SF program is scheduled to be called)
b	27/28, 4/5	Contains SF program number (if an SF program is scheduled to be called)
c	28/29, 3.4	Contains SF program number (if an SF program is scheduled to be called)
d	29/30, 2/3	Contains SF program number (if an SF program is scheduled to be called)
e	30/31, 1/2	Contains SF program number (if an SF program is scheduled to be called)
f	31/32, 0/1	Contains SF program number (if an SF program is scheduled to be called)

Note: If the bit is set to 1, the option is enabled.

LRSF Bit Definitions

Ramp/Soak Status Data Element Format

MSB		LSB	
ABCD	EFGH	IJKL	MNOP

The LRSF bit has the following format:

V/W, X/Y

V=0-based bit addressing, bit 0 is MSB

W=1-based bit addressing, bit 1 is MSB (Simatic 505 Documentation)

X=0-based bit addressing, bit 0 is LSB (Server Default Setting)

Y=1-based bit addressing, bit 1 is LSB

Element	LRSF bit	Description
A	0/1, 15/16	1=Restart at the first step. To restart, toggle bit off, on, then off again. The restart occurs on the trailing edge of a square wave.
B	1/2, 14/15	1=Hold at the current step. To hold, set bit on.
C	2/3, 13/14	1=Jog to next step. To jog, set bit on. Jog occurs on the rising edge of a square wave.
D	3/4, 12/13	1=Finish. Indicates ramp/soak is completed.
E	4/5, 11/12	1=Wait. This bit is set during a soak period when the PV is not within a specified

		deviation from the SP. The loop holds the soak timer when bit 5 is set.
F	5/6, 10 /11	1=Hold in progress at current step
G	6/7, 9 /10	Unused (always returned as 0)
H	7/8, 8 /9	Unused (always returned as 0)
I	8/9, 7 /8	Ramp/soak step number currently executing
J	9/10, 6 /7	Ramp/soak step number currently executing
K	10/11, 5 /6	Ramp/soak step number currently executing
L	11/12, 4 /5	Ramp/soak step number currently executing
M	12/13, 3 /4	Ramp/soak step number currently executing
N	13/14, 2 /3	Ramp/soak step number currently executing
O	14/15, 1 /2	Ramp/soak step number currently executing
P	15/16, 0 /1	Ramp/soak step number currently executing

Find Forced Addressing

Address Type	Format	Range	Data Types	Access
Find Forced Discrete Starting at X address	FFX	N/A	String	Read Only
Find Forced Discrete Starting at Y address	FFY	N/A	String	Read Only
Find Forced Discrete Starting at C address	FFC	N/A	String	Read Only
Find Forced Word Starting at WX address	FFWX	N/A	String	Read Only
Find Forced Word Starting at WY address	FFWY	N/A	String	Read Only
Find Forced Discrete/Word Starting address	FFSTART	N/A	DWord	Read/Write

Note 1: Each only allows the one data type stated and none have a range.

Note 2: FFSTART is a driver internal tag (one per device) that is initialized to 1 on server startup. It has a write range of 1->65536 and a write of values outside of this range will fail with an error message.

Note 3: If the starting address value in FFSTART is greater than the range for the Find Forced type issued, then the read will fail with an "Address out of Range" error (Task Code error '02') and the quality of the tag will be set to "Bad". For example: If FFSTART was set to a value of 65536 and a read of "FFC" was issued, the error generated would look similar to the following due to C memory in the PLC only being configured for maximum value of 32768: "Device 'Channel1.Device1' detected a Task Code error '02' (Tag 'ffc', Size '256')."

Usage

1. Perform a Write to FFSTART with the value of the starting address to be used for the Find Forced type in Step 2.

Caution: Make sure that the value written is within the configured range for the type.

2. Perform a Read using the Find Forced type selected. The read issues a command to the PLC asking it to perform a search for any Forced discretes (C, X/Y) or words (WX/WY).

Find Forced Search Algorithm

Using the Start Address provided, the first forced address beginning with or following the starting address is output. The search will continue until it completes. A search is completed by one of the following: (1) the search proceeds through all locations in a circular manner and arrives back at the specified starting address or (2) enough addresses are found to fill the output buffer. The search will cycle through all address types (C then X/Y for discretes, WX/WY for words) for the command given.

If the search responds with the maximum number of addresses, then another request, beginning with the address following the last forced address returned, must be issued to continue the search. This must be repeated until the original starting address is reached to produce the complete list if a large number of addresses are forced.

Find Forced Search Notes

1. X/Y points reside in the same memory space (i.e. X1=Y1, X2=Y2, etc). The PLC will preferentially express X/Y points using Y addresses unless the search is explicitly started using an X address. Thus, using FFX versus FFY lets users

choose whether to have the results returned using an "X:address" format or a "Y:address" format.

2. WX/WY words reside in the same memory space (i.e. WX1=WY1, WX2=WY2, etc). The PLC will preferentially express WX/WY words using WY addresses unless the search is explicitly started using a WX address. Thus, using FFWX versus FFWY lets users choose whether to have the results returned using an "WX:address" format or a "WY:address" format.

Example

1. Write a value of 600 to a tag named 'ffstart', with an address of FFSTART.

2. Read a tag named 'ffc', with an address of FFC. Value returned in datatype of string: "<Start Address 00600>C:00600,C:00601,Y:00001,Y:01024,C:00001,C:00002"

This example illustrates the algorithm by which the Find Forced search is performed. The search was started at address C:00600 and it found two discretes forced at addresses C:00600,C:00601. It then proceeded to search X/Y discretes where it found Y:00001,Y:01024 forced. Next, it looped back around to the start of C discretes and found C:00001, C:00002 forced. Finally, it ended its search at C:00599 without finding any further forced discretes.

Output Formats

1. None (No Forced addresses found).
2. <Start Address SSSSS>Type1:Address1... up to ... Type15:Address15

where

SSSSS: value of FFSTART

Type: one of: C, X, Y, WX, WY

Address: fixed at five characters (00001 -> 65536)

Notes:

1. Minimum of 1 address returned.
2. Maximum of 15 addresses returned if each returned address uses a one word format.
3. If an address consumes two words (due to the address not being within the first page), then the maximum output will be reduced by one for each two-word address. Users can determine if an address will consume one word or two by using the following:

C - Addresses

1 through 512 = one word

Addresses 513 through 65536 = two words

X,Y,WX,WY - Addresses

1 through 1024 = one word

Addresses 1025 through 65536 = two words

4. Maximum of 7 addresses returned if each returned address consumes two words.
5. Final output is independent of the address being returned in a one word or two word format.
6. Maximum string length is internally set to 256.

Error Descriptions

The following error/warning messages may be generated. Click on the link for a description of the message.

Address Validation

[Missing address](#)

[Device address '<address>' contains a syntax error](#)

[Address '<address>' is out of range for the specified device or register](#)

[Data Type '<type>' is not valid for device address '<address>'](#)

[Device address '<address>' is Read Only](#)

[Array size is out of range for address '<address>'](#)

[Array support is not available for the specified address: '<address>'](#)

Device Status Messages

[Device '<device name>' is not responding](#)

[Unable to write to '<address>' on device '<device name>'](#)

Driver Error Messages

[Winsock initialization failed \(OS Error = n\)](#)

[Winsock V1.1 or higher must be installed to use the Simatic 505 Ethernet device driver](#)

[Device '<device name>' input queue is full. The module is receiving requests faster than it can process requests](#)

[Device '<device name>' cannot execute a read request \('<address>', '<size>'\)](#)

[Device '<device name>' cannot execute a write request \('<address>', '<size>'\)](#)

[Device '<device name>' responded with Extended error '<error>' \(Tag '<address>', Size '<size>'\)](#)

[Device '<device name>' detected a NITP Protocol Error \(Tag '<address>', Size '<size>'\)](#)

[Device '<device name>' detected a Task Code error '<error>' \(Tag '<address>', Size '<size>'\)](#)

[Device '<device name>', Write Value not in range 1-->65536. \(Tag 'FFSTART'\)](#)

Address Validation

The following error/warning messages may be generated. Click on the link for a description of the message.

Address Validation

[Missing address](#)

[Device address '<address>' contains a syntax error](#)

[Address '<address>' is out of range for the specified device or register](#)

[Data Type '<type>' is not valid for device address '<address>'](#)

[Device address '<address>' is Read Only](#)

[Array size is out of range for address '<address>'](#)

[Array support is not available for the specified address: '<address>'](#)

Missing address

Error Type:

Warning

Possible Cause:

A tag address that has been specified dynamically has no length.

Solution:

Re-enter the address in the client application.

Device address '<address>' contains a syntax error

Error Type:

Warning

Possible Cause:

A tag address that has been specified dynamically contains one or more invalid characters.

Solution:

Re-enter the address in the client application.

Address '<address>' is out of range for the specified device or register

Error Type:

Warning

Possible Cause:

A tag address that has been specified dynamically references a location that is beyond the range of supported locations for the device.

Solution:

Verify the address is correct; if it is not, re-enter it in the client application.

Data Type '<type>' is not valid for device address '<address>'**Error Type:**

Warning

Possible Cause:

A tag address that has been specified dynamically has been assigned an invalid data type.

Solution:

Modify the requested data type in the client application.

Device address '<address>' is Read Only**Error Type:**

Warning

Possible Cause:

A tag address that has been specified dynamically has a requested access mode that is not compatible with what the device supports for that address.

Solution:

Change the access mode in the client application.

Array size is out of range for address '<address>'**Error Type:**

Warning

Possible Cause:

A tag address that has been specified dynamically is requesting an array size that is too large for the address type or block size of the driver.

Solution:

Re-enter the address in the client application to specify a smaller value for the array or a different starting point.

Array Support is not available for the specified address: '<address>'**Error Type:**

Warning

Possible Cause:

A tag address that has been specified dynamically contains an array reference for an address type that doesn't support arrays.

Solution:

Re-enter the address in the client application to remove the array reference or correct the address type.

Device Status Messages

The following error/warning messages may be generated. Click on the link for a description of the message.

Device Status Messages

[Device '<device name>' is not responding](#)

[Unable to write to '<address>' on device '<device name>'](#)

Device '<device name>' is not responding

Error Type:

Serious

Possible Cause:

1. The connection between the device and the host PC is broken.
2. The IP address assigned to the device is incorrect.

Solution:

1. Verify the cabling between the PC and the PLC device.
2. Verify that the IP address given to the named device matches that of the actual device.

Unable to write to '<address>' on device '<device name>'

Error Type:

Serious

Possible Cause:

1. The connection between the device and the host PC is broken.
2. The named device may have been assigned an incorrect IP address.

Solution:

1. Verify the cabling between the PC and the PLC device.
2. Verify that the IP address given to the named device matches that of the actual device.

Driver Error Messages

The following error/warning messages may be generated. Click on the link for a description of the message.

Driver Error Messages

[Winsock initialization failed \(OS Error = n\)](#)

[Winsock V1.1 or higher must be installed to use the Simatic 505 Ethernet device driver](#)

[Device '<device name>' input queue is full. The module is receiving requests faster than it can process requests](#)

[Device '<device name>' cannot execute a read request \('<address>', '<size>'\)](#)

[Device '<device name>' cannot execute a write request \('<address>', '<size>'\)](#)

[Device '<device name>' responded with Extended error '<error>' \(Tag '<address>', Size '<size>'\)](#)

[Device '<device name>' detected a NITP Protocol Error \(Tag '<address>', Size '<size>'\)](#)

[Device '<device name>' detected a Task Code error '<error>' \(Tag '<address>', Size '<size>'\)](#)

[Device '<device name>', Write Value not in range 1-->65536. \(Tag 'FFSTART'\)](#)

Winsock initialization failed (OS Error = n)

Error Type:

Fatal

OS Error Indication	Possible Solution
10091	Indicates that the underlying network subsystem is not

	ready for network communication. Wait a few seconds and restart the driver.
10067	Limit on the number of tasks supported by the Windows Sockets implementation has been reached. Close one or more applications that may be using Winsock and restart the driver.

Winsock V1.1 or higher must be installed to use the Simatic 505 Ethernet device driver

Error Type:

Fatal

Possible Cause:

The version number of the Winsock DLL found on the system is less than 1.1.

Solution:

Upgrade Winsock to version 1.1 or higher.

Device '<device name>' input queue is full. The module is receiving requests faster than it can process requests

Error Type:

Serious

Possible Cause:

The device is handling requests from more than one PC at a time.

Solution:

Using the device programming software, make sure that the "task codes per scan time" is set to 8. Adjusting the scan time can also improve device performance. It is recommended to set the scan time to Variable. Consult the device documentation for more details on adjusting these settings.

Device '<device name>' cannot execute a read request ('<address>', '<size>')

Error Type:

Serious

Possible Cause:

The device is unable to return the amount of read data requested in one request.

Solution:

Adjust the requested block size on the device so that the driver makes smaller requests.

Device '<device name>' cannot execute a write request ('<address>', '<size>')

Error Type:

Serious

Possible Cause:

The device is unable to receive the amount of writer data sent in one request.

Solution:

Adjust the array size on the address so that a smaller amount of data is sent to the device in one transaction.

Device '<device name>' responded with Extended error '<error>' (Tag '<address>', Size '<size>')

Error Type:

Informational

Possible Cause:

See Error message prior to this one.

Details:

This error is used for extended error information by some error responses.

Device '<device name>' detected a NITP Protocol Error (Tag '<address>', Size '<size>')

Error Type:

Serious

Possible Cause:

This error may occur when sending Packed Task Code messages. The device is experiencing network errors.

Solution:

Check Network connections.

Device '<device name>' detected a Task Code error '<error>' (Tag '<address>', Size '<size>')

Error Type:

Serious

Possible Cause:

This error may occur when sending Packed Task Code messages.
The device is experiencing network errors or PLC encountered an error when processing a task code.

Solution:

1. Check Network connections.
2. Refer to the specific error code.

See Also:

[Task Error Codes](#)

Device '<device name>', Write Value not in range 1-->65536. (Tag 'FFSTART')

Error Type:

Warning

Possible Cause:

Value written outside of range specified.

Solution:

Write a value in the range of 1-65536.

Task Code Error Codes

Code	Description
02	Address out of Range (Other than Ladder Logic)
03	Requested Data not Found
04	Illegal Task Code Request (e.g. Task Code not Supported)

05	Request Exceeds Program Memory Size (Ladder Logic)
07	Fatal Error Detected
09	Incorrect amount of Data sent with Request
0C	Attempted Write Operation Did Not Verify
0D	Illegal Number of ASCII Characters Received
0E	Illegal Write to Program Memory (Non Volatile)
11	Invalid Data sent with the Command
16	Attempted Write to a Protected Variable (e.g. TCC, TCP)
17	No response from PLC (e.g. Single Scan not performed)
18	Requested memory size exceeds total available memory
19	Requested Memory size is not a multiple of block allocation size
1A	Requested memory size is less than minimum defined value.
1B	Requested memory size is larger than maximum defined value
1C	PLC Busy - Cannot complete the requested operation
3F	Bus Error Detected
40	Operating System Error Detected
4A	Attempt to access an integer only variable as a real.
4B	Attempt to access a real -only value as an integer.
4C	Task Code buffer overflow -- too much data requested
4E	Attempt to write a read only variable.
4F	Invalid variable type for this operation
50	Task code request buffer too large (PLC internal error)

Index

- 5 -

505 Protocol 5
 505-CP1434-TCP Addressing 9
 505-CP2572 Addressing 9

- A -

ACF Bit Definitions 15
 Address '<address>' is out of range for the specified device or register 23
 Address Descriptions 9
 Address Validation 23
 addressing
 CP1434 9
 CP2572 9
 Addressing Options 4
 Alarm Addressing 13
 Array size is out of range for address '<address>' 24
 Array support is not available for the specified address
 '<address>' 24
 AVF Bit Definitions 15

- B -

BCD 8
 Boolean 8

- C -

Cable Diagrams 6
 CAMP 5
 Common Data Type Addressing 9
 Communications Parameters 5
 Connection Limitations 3
 CP1434 9
 CP2572 9

- D -

Data Type '<type>' is not valid for device address '<address>' 24
 Data Types Description 8
 Device '<device name>'
 Write Value not in range1-->65536. (Tag 'FFSTART') 27
 Device '<device name>' cannot execute a read request (<address> <size>) 26
 Device '<device name>' cannot execute a write request (<address> <size>) 26
 Device '<device name>' input queue is full. The module is receiving requests faster than it can process requests 26
 Device '<device name>' is not responding 25
 Device address '<address>' contains a syntax error 23
 Device address '<address>' is read only 24
 Device ID 3
 Device Status Messages 25
 Driver Error Messages 25
 DWord 8

- E -

Error Descriptions 22

- F -

Find Forced Addressing 21
 Float 8

- L -

LBCD 8
 LCF Bit Definitions 19
 Long 8
 Loop Addressing 16
 LRSF Bit Definitions 20
 LVF Bit Definitions 18

- M -

Missing address 23

- N -

Network 3

NITP 5

Number of connections 3

- O -

Optimizing Simatic505 Ethernet Communications
7

Overview 3

- P -

Packed Task Code 5

- S -

Short 8

- T -

Task Code Error Codes 27

TISOFT 5

- U -

Unable to write tag '<address>' on device '<device
name>' 25

- W -

Winsock initialization failed (OS Error = n) 25

Winsock V1.1 or higher must be installed to use
the Simatic 505 Ethernet device driver 26

Word 8